

Übungsblatt 0/A¹

Lernziele: Starten und Stoppen des Arbeitsplatzrechners, Anmelden, Abmelden, Arbeiten mit der Linux Shell, Umgang mit dem elearning-Server der Frankfurt University of Applied Sciences.

1 Die Linux Shell

Die Rechner in den Informatik-Laboren im 2. OG., Gebäude 1, sind mit Microsoft Windows und einem Linux-Betriebssystem (openSUSE) ausgestattet. Nach Einschalten des Rechners kann mit Hilfe der Pfeiltasten das gewünschte Betriebssystem ausgewählt werden.

In den Übungen wird mit dem Linux-Betriebssystem gearbeitet. Zur Kommunikation mit dem System wird die traditionelle kommandozeilenorientierte Benutzerschnittstelle, die “Unix-Shell“, verwendet.

Die Shell ist ein Kommandozeileninterpreter: Gibt ein Benutzer ein Kommando in der Befehlszeile ein, so wird dieses ausgewertet und, falls kein Syntaxfehler gefunden wurde, sofort ausgeführt.

Es gibt verschiedene Shell-Programme, die sich in Funktionsumfang und Syntax unterscheiden. In den Übungen wird die Bourne-Again-Shell (bash) verwendet. Die Shell ist ein sehr mächtiges Werkzeug, das Ihnen wesentlich mehr Möglichkeiten bietet, als wir in den Programmierübungen benötigen werden:

Kommandos können auf verschiedene Weise miteinander verknüpft werden. Die Shell stellt Kontrollstrukturen vergleichbar einer höheren Programmiersprache zur Verfügung, mit deren Hilfe aus den Kommandos komplexe Prozeduren, sogenannte Shellskripte, erstellt werden können. Diese können genauso wie ein Kommando durch Eingabe ihres Namens gestartet werden.

1.1 Die vollständige Befehlszeile

Shell-Kommandos haben grundsätzlich den folgenden Aufbau:

`Kommandoname [Optionen] [Argumente]`

Optionen sind an dem vorangestellten - Zeichen zu erkennen. Die eckigen Klammern deuten an, dass die Angabe von Optionen und Argumenten nicht bei allen Kommandos zwingend notwendig ist.

In Linux wird im Gegensatz zu Windows immer zwischen Groß- und Kleinschreibung unterschieden. Das Nichtberücksichtigen von Groß- und Kleinschreibung führt zu einer Fehlermeldung des Systems.

Eine Fehlermeldung wie

```
bash: Gcc: command not found
```

¹Mit Dank an Frau Gabriele Czermin für die ursprüngliche Version dieses Dokuments.

könnte durch eine falsche Schreibweise des Kommandos ausgelöst worden sein.

1.2 Dateien

Die Speicherung von Daten geschieht in Dateien (englisch: files), die unter Unix einfach aus aufeinanderfolgenden Zeichen bestehen. Dateien haben Namen, unter denen sie mit Hilfe von Shell-Kommandos oder Programmen manipuliert werden können.

Namen von Dateien sind wiederum in Dateien abgespeichert, den sogenannten Verzeichnissen (englisch: directories). Das aus Dateien bzw. Verzeichnissen bestehende Teilsystem einer Computeranlage heißt Dateisystem (englisch: file system)

Das Dateisystem von **Unix** ist (wie das von den MS Windows Betriebssystemen) hierarchisch strukturiert.

Es gibt in diesem System

- Verzeichnisse (directories) z.B. `/home`
- gewöhnliche Dateien (files) z.B. `termin.txt`
- spezielle Dateien (devices, special files), die physikalische oder logische Geräte bezeichnen z.B. `/dev/sr0` (das erste SCSI CD-Laufwerk) oder `/dev/pts/2` (das Bildschirmfenster oder **P**seudoterminal mit der Nummer 2).

Ein Verzeichnis kann dabei Dateien oder auch wieder Verzeichnisse enthalten; solche Verzeichnisse heißen dann Unterverzeichnisse der darüberliegenden. Dabei gibt es ein Verzeichnis, so dass alle übrigen Verzeichnisse Unterverzeichnisse dieses sogenannten **root**-Verzeichnisses sind. Dieses **root**-Verzeichnis hat den Unix-Namen `/`. (In den MS Windows Betriebssystemen lautet dieser Name `<Laufwerksbuchstabe>:\` z.B. `C:\`.) Die Verzeichnisse mit den Unterverzeichnissen bilden eine baumartige Struktur mit der Wurzel im **root**-Verzeichnis und mit Dateien als Blätter.

In Unix gibt es zwei Namen für Verzeichnisse, die eine spezielle Bedeutung haben:

- `“.”` gibt das aktuelle Verzeichnis an, also das Verzeichnis, in dem sich der Benutzer in der aktuellen Shell-Sitzung gerade befindet.
- `“..”` gibt das unmittelbar über dem aktuellen Verzeichnis befindliche Verzeichnis an. Es ist also `“.”` unmittelbares Unterverzeichnis von `“..”`.

Jede Benutzerin und jeder Benutzer besitzt ein sogenanntes Heimatverzeichnis (home directory), in dem er sich unmittelbar nach dem Öffnen eines Shellfensters befindet. Dieses Heimatverzeichnis kann stets auch über die Zeichenreihe `$HOME` oder das Zeichen `~` angesprochen werden.

Beispiel für das Heimatverzeichnis der Studentin Überfliegerin im Studiengang Informatik - mobile Anwendungen:

`/home/student/mobil/ueberfliegerin.`

1.3 Absolute und relative Pfadnamen

Beim Navigieren durch das Verzeichnissystem können absolute oder relative Pfadnamen verwendet werden.

Der absolute Pfadname besteht aus der vollständigen Liste aller Verzeichnisse, die passiert werden müssen, um vom root-Verzeichnis aus zur gewünschten Datei zu gelangen, jeweils durch / getrennt.

Beispiele für absolute Pfadnamen:

```
/etc
```

```
/home/student/temp/temp_14
```

```
/usr/include/crypt.h
```

Relative Pfadnamen bestehen aus allen Verzeichnissen (jeweils durch / getrennt), die durchlaufen werden müssen, um vom aktuellen Verzeichnis aus zur gewünschten Datei zu gelangen. Beispiele für relative Pfadnamen vom Verzeichnis `/home/student/temp` aus gesehen:

```
temp_23
```

```
temp_2/.profile
```

```
../mobil
```

```
../../../../usr
```

1.4 Die Manual Pages, Kurzbeschreibung der Kommandos, KDE Help Center

Das Linux-System enthält ein sehr umfangreiches Online-Hilfesystem, die so genannten Manual-Pages. Für nahezu jedes Kommando existiert eine Beschreibung, die mit dem Befehl `man <Kommando>` ausgegeben werden kann. (Ausdrücke der Form `<Text>` werden Ihnen noch öfter begegnen. Sie sind Platzhalter für verschiedene Möglichkeiten von konkreten Ausdrücken und müssen beim Anwenden ersetzt werden. Hier steht `<Kommando>` für ein beliebiges Shell-Kommando. `man <Kommando>` könnte beispielsweise durch `man ls` oder `man mkdir` ersetzt werden.)

Der Befehl `whatis <Kommando>` gibt eine Kurzbeschreibung des Kommandos aus.

Die Manual-Pages sind nach Themen sortiert, z.B. enthält der Abschnitt 1 die Beschreibungen der ausführbaren Programme und Shellbefehle, der Abschnitt 3 C-Bibliotheksfunktionen.

Eine Manual-Page zu einem Kommando ist in Unterpunkte gegliedert. Üblicherweise enthält eine Seite die Unterpunkte **SYNOPSIS** (Beschreibung der Syntax), **DESCRIPTION** (Kurzbeschreibung des Kommandos) **OPTIONS** (Optionen). Dazu können noch andere Unterpunkte wie **FILES** (Dateien), **SEE ALSO** (verwandte Kommandos), **DIAGNOSTICS** (Fehlermeldungen, die bei der Ausführung erzeugt werden können), **EXAMPLE** (Beispiel zum Aufruf des Kommandos) vorhanden sein.

Eine Manual-Page wird seitenweise auf dem Bildschirm ausgegeben. Durch Betätigen der Leertaste wird die Ausgabe fortgesetzt, mit dem Buchstaben **q** wird die Ausgabe abgebrochen.

Mit `</Begriff>` wird die Manual-Page ab der aktuellen Position vorwärts nach dem angegebenen Begriff durchsucht. Der Cursor markiert das erste Auftreten des gesuchten Begriffes. Mit **n** wird vorwärts, mit **N** rückwärts nach dem nächsten Auftreten des Begriffes gesucht.

Zu vielen Befehlen existiert außerdem eine Kurzbeschreibung, welche mit `<kommando> --help` eingesehen werden kann.

Das "KDE Help Center" ist ein Werkzeug, das die verschiedenen in dem System vorhandenen Dokumente zu Linux und verwandten Themen in übersichtlicher Form und einer ansprechenden Formatierung darstellt. Auch die Manual-Pages finden sich in diesem Programm wieder. Sie sind mit einem Index versehen und innerhalb der einzelnen Abschnitte alphabetisch sortiert. Sie können allerdings nicht mit dem oben beschriebenen Mechanismus durchsucht werden.

Das Hilfesystem kann aus dem **K-Menü** gestartet werden. Geben Sie in dem Feld mit der Beschriftung **Search...** den Begriff **help** ein. Das Startsymbol für das Hilfesystem ist ein Rettungsring.

2 Die wichtigsten Kommandos und Programme für die Übungen

Grundlagen rund um Linux finden Sie z.B. unter <http://de.linwiki.org/wiki/Linuxfibel>. Für alle, die sich eingehender mit Linux beschäftigen möchten, empfehle ich das Tutorial <https://linuxjourney.com/>. Die Sprache kann auf "Deutsch" umgeschaltet werden.

2.1 Einführende Aufgaben

1. Schalten Sie den Rechner und den Monitor an Ihrem Arbeitsplatz ein und starten Sie das Betriebssystem Linux.
2. Melden Sie sich mit der Benutzerkennung und dem Passwort an, die Sie von der Abteilung Campus IT erhalten haben. Beim ersten Anmelden eines neuen Benutzers wird ein Fenster mit Informationen auf der Arbeitsfläche geöffnet. Bewegen Sie den Mauszeiger auf die Schaltfläche **Schließen** und betätigen Sie die linke Maustaste. Schließen Sie ebenfalls das Fenster mit den Tipps.
3. Schauen Sie sich die Kontrollleiste am unteren Bildschirmrand an. Klicken Sie auf das Symbol (Chamäleonkopf) ganz links auf der Kontrollleiste und machen Sie sich mit dem "K-Menü" vertraut.
4. Starten Sie einen Webbrowser. Welchen Browser haben Sie gefunden?
5. Rufen Sie die Startseite der Frankfurt University of Applied Sciences auf (Link: <http://www.frankfurt-university.de>).

6. Suchen Sie die Homepage der Abteilung Campus IT auf. Falls Sie noch keine Benutzerkennung haben: Verfahren Sie wie unter “DV-Benutzerkennung für Internetdienste“ beschrieben und vergessen Sie nicht, in den nächsten Tagen Ihre Benutzerkennung und Ihr Passwort bei der Benutzerberatung von Campus IT abzuholen!
7. Geben Sie im Adressfeld des Browserfensters die Adresse `https://campuas.frankfurt-university.de` ein. Schreiben Sie sich in die Kurse
 - “Informationen der Studiengangsleitungen I, WI und HIS”,
 - “Sertkaya: Einführung in die Programmierung mit C“ und
 - “Informatik, Informatik - mobile Anwendungen - Infos für den Start ins Studium”

ein. Die benötigten Zugangsschlüssel erhalten Sie in der Übung. Melden Sie sich vom elearning-Server ab und beenden Sie das Browser-Programm.

8. Starten Sie das Programm **Konqueror**. Klicken Sie auf das Symbol für Ihr Heimatverzeichnis unterhalb des Pfeils neben dem Home-Page-Symbol(Haus) in der Menü-Leiste. Schreiben Sie den Namen Ihres Heimatverzeichnisses aus dem Adressfeld ab. Aktivieren Sie **Show Hidden Files** unterhalb des Menüpunkts **View**. Notieren Sie die Namen von mindestens 2 Verzeichnissen und 2 Dateien in Ihrem Heimatverzeichnis. Verschaffen Sie sich einen Überblick über die Funktionalität des Programms Konqueror. Probieren Sie verschiedene Funktionen des Programms Konqueror aus, aber löschen und verschieben Sie dabei keine Dateien oder Ordner! Beenden Sie das Programm.

2.2 Verwendung von Shell-Kommandos - Aufgaben

Die folgenden Aufgaben lassen sich mit Hilfe der Befehle aus der Tabelle ?? lösen. Allerdings werden Sie für die Lösung der Aufgaben immer wieder die Online-Manual-Pages oder die Kurzbeschreibung(<**kommando**> --help) zu Rate ziehen müssen.

1. Öffnen Sie ein “Shell-Fenster“ indem Sie das Programm **konsole** aus dem K-Menü starten. Die weiteren Aufgaben werden auf der Kommandozeile dieses Fensters ausgeführt.
2. Erzeugen Sie in Ihrem Heimatverzeichnis ein Unterverzeichnis namens **C-Uebungen**. Wie lauten der relative Pfadname (von Ihrem Heimatverzeichnis aus gesehen) und der absolute Pfadname dieses Verzeichnisses?

Wechseln Sie in das neu angelegte Verzeichnis. Überprüfen Sie mit dem Kommando **pwd**, ob der Wechsel in das Verzeichnis erfolgreich durchgeführt wurde.

Bemerkung: Der Ordner **C-Uebungen** ist das Arbeitsverzeichnis für unsere Übung. Sorgen Sie dafür, dass alle Dateien und Verzeichnisse, die Sie im Laufe des Semesters erzeugen, unterhalb dieses Ordners abgelegt werden.

3. Erzeugen Sie in dem neuen Verzeichnis eine Datei mit dem Editor `kate`: Überlegen Sie sich einen Dateinamen. **Bitte verwenden Sie prinzipiell keine Leerzeichen und Umlaute in Unix-Dateinamen.** Geben Sie das Kommando `kate <Dateiname>&` ein. Dadurch erhält die neue Datei gleich beim Anlegen den von Ihnen gewählten Namen. Das Kaufmanns- und `&` bewirkt, dass der Editor “im Hintergrund” gestartet wird und das Fenster, in dem er aufgerufen wurde, wieder für Eingaben freigegeben wird. Tippen Sie einige Zeilen in die neue Datei. Speichern Sie die Datei und beenden Sie das Programm `kate`.
4. Überprüfen Sie mit Hilfe des Befehls `ls`, ob die neue Datei erzeugt wurde. Geben Sie den Inhalt der Datei mit dem Kommando `cat` aus. Löschen Sie die Datei mit dem Kommando `rm`.
5. Kopieren Sie mit dem Kommando `cp` die Datei `bitte_fehler_beheben.c` aus dem Ordner
`/home/prof/sertkaya/c-uebung/vorlage` in den Ordner `C-Uebungen`. Überprüfen Sie, ob die Datei in Ihrem Verzeichnis `C-Uebungen` vorhanden ist. Wechseln Sie in Ihr Heimatverzeichnis.
6. Melden Sie sich vom System ab. Nach dem erfolgreichen Abmelden vom System erscheint erneut das Anmeldefenster. Suchen Sie in diesem Fenster den Button zum Ausschalten des Rechners und betätigen Sie diesen mit der linken Maustaste. Schalten Sie den Monitor aus.

| Befehl / Programm | Beispiele | Beschreibung |
|------------------------------|--|---|
| man | man ls | manual pager - Linux Handbuchseiten |
| ls | ls -l | list - Inhalt eines Verzeichnisses ausgeben, ohne Argument wird das aktuelle Verzeichnis ausgegeben |
| cd | cd /home/student cd.. cd | change directory - Verzeichnis wechseln |
| pwd | pwd | print name of working directory - Namen des aktuellen Verzeichnisses anzeigen |
| cat | cat /home/prof/sertkaya/c-uebung/Shell-Tipps.txt | concatenate - Inhalt von Dateien auf dem Bildschirm ausgeben |
| more | more /usr/include/limits.h | seitenweise Ausgabe auf dem Bildschirm |
| less | less /etc/profile | seitenweise Ausgabe auf dem Bildschirm |
| mkdir | mkdir C-Programme | make directory - Verzeichnis anlegen |
| cp | cp /home/prof/sertkaya/c-uebung/vorlage/bitte_fehler_beseitigen.c beispiel.c cp beispiel.c Sicherheitskopien | copy - Dateien kopieren |
| mv | mv beispiel.c programm_mit_fehlern.c mv programm_mit_fehlern.c C-Programme | move - Dateien umbenennen bzw. im Verzeichnisbaum verschieben |
| rm | rm beispiel.c | remove - Datei unwiderruflich löschen |
| rmdir | rmdir alt | remove directory - leeres Verzeichnis unwiderruflich löschen |
| gcc | gcc -o helloworld helloworld.c | GNU C Compiler: Erzeugen eines ausführbaren Programms aus dem C-Quellprogramm |
| kate | kate helloworld.c | Texteditor : Programm zum Erstellen und Bearbeiten von Dateien |
| vim | vim helloworld.c | Texteditor (für fortgeschrittene Shell-Benutzer) |
| firefox | | Webbrowser |
| konqueror | | Webbrowser und Dateimanager für die grafische Linux-Benutzeroberfläche KDE |